

## MODUL PRAKTIKUM VI

### INHERITANCE

#### A. Tujuan

1. Mahasiswa mampu menerapkan dan menganalisa konsep *inheritance* pada program Java
2. Mahasiswa mampu menerapkan dan menganalisa penggunaan *overriding method* pada program Java
3. Mahasiswa mampu menerapkan dan menganalisa penggunaan *keyword super* pada program Java.

#### B. Latihan

##### Latihan 1:

Implementasikan *source code* berikut :

##### 1. Person.java

```
class Person
{
    protected String name;
    protected int age;

    public Person(String name, int age)
    {
        this.name = name;
        this.age = age;
    }

    //metode
    public void info()
    {
        System.out.println ("Nama : "+this.name);
        System.out.println ("Usia : "+this.age);
    }
    //akhir kelas program
}
```

##### 2. Employ.java

```
class Employ extends Person
{
    private String noKaryawan;

    //konstruktor
    public Employ(String noKaryawan, String name, int age)
    {
        super (name, age);
        this.noKaryawan = noKaryawan;
    }

    //metode
    public void info ()
    {
        System.out.println ("No.karyawan :"+this.noKaryawan);
        super.info ();
    }
}
```

### 3. KonstruktorSuperKelas.java

```
public class KonstruktorSuperKelas
{
    public static void main (String [] args)
    {
        Employ programmer1 = new Employ ("12345678", "Yanto", 32);
        programmer1.info();
    }
}
```

a. Output dari program di atas adalah:

b. Tuliskan UML class diagram dari *class* Person.java dan Employ.java beserta relasinya:

### Latihan 2:

Implementasikan *source code* berikut :

#### 1. A.java

```
class A
{
    private int a;
    public void setA (int nilai)
    {
        a = nilai;
    }

    public int getA()
    {
        return a;
    }

    public void tampilkanNilai()
    {
        System.out.println ("Nilai a : "+getA());
    }
}
```

## 2. B.java

```
class B extends A
{
    private int b;
    public void setB (int nilai)
    {
        b = nilai;
    }

    public int getB ()
    {
        return b;
    }

    //melakukan override terhadap method tampilkanNilai ()
    //yang terdapat pada kelas A

    public void tampilkanNilai ()
    {
        super.tampilkanNilai (); //memanggil method dalam kelas A
        System.out.println ("Nilai b : "+getB());
    }
}
```

## 3. DemoOverride.java

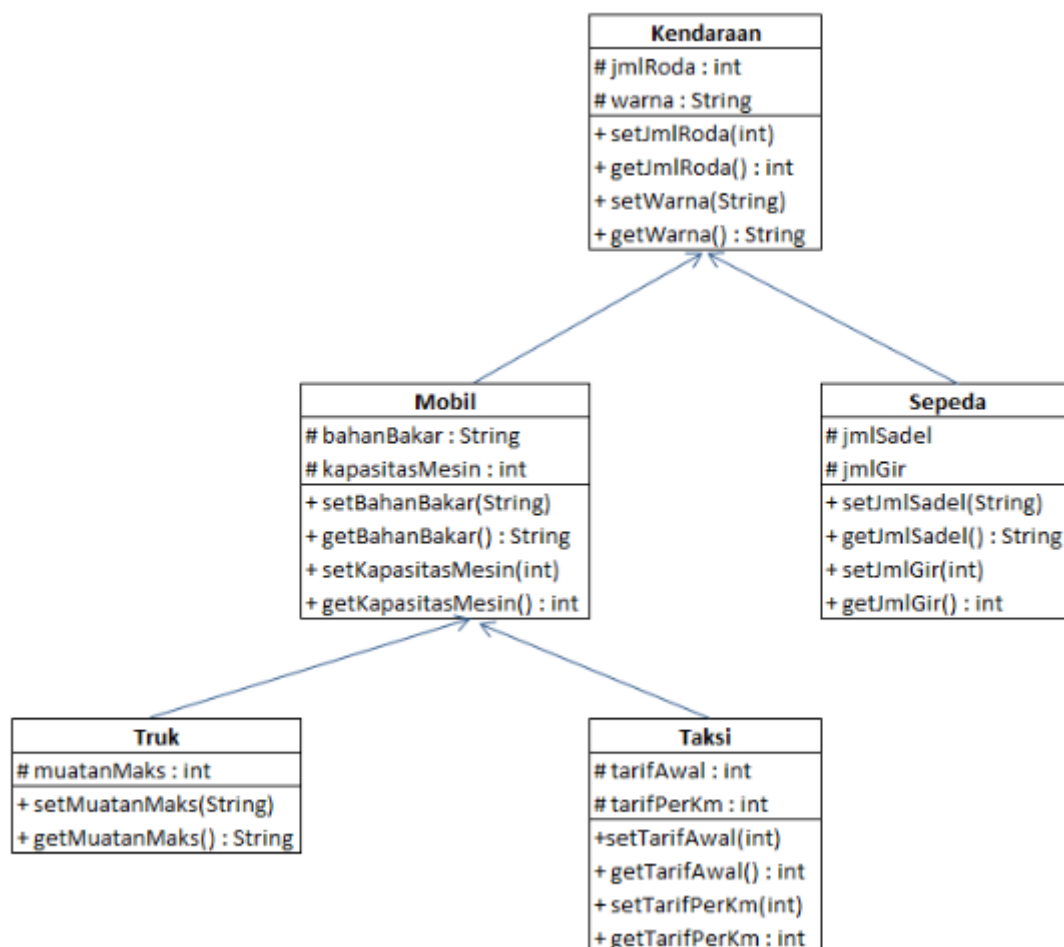
```
class DemoOverride
{
    public static void main (String [] args)
    {
        B obj = new B ();
        obj.setA(50);
        obj.setB(150);
        //akan memanggil method yang terdapat pada kelas B
        obj.tampilkanNilai ();
    }
}
```

a. Output dari program di atas adalah:

b. Tuliskan UML class diagram dari *class* A.java dan B.java beserta relasinya:

### C. Tugas Praktikum

Implementasikan UML *class diagram* berikut dalam program Java



Selanjutnya buatlah kelas Tes.java yang membuat obyek-obyek serta mengeset nilai variabel seperti pada tabel berikut:

Obyek	jmlRoda	warna	bahanBakar	kapasitasMesin	muatanMaks	
truk1	4	Kuning	solar	1500	1000	
truk2	6	merah	Solar	2000	5000	
					tarifAwal	tarifPerKm
taksi1	4	oranye	bensin	1500	10000	5000
taksi2	4	biru	bensin	1300	7000	3500
			jmlSadel	jmlGir		
sepeda1	3	Hitam	1	2		
sepeda2	2	Putih	2	5		

#### **D. Tugas Rumah**

Buat UML *class diagram* dan program Java yang mengimplementasikan konsep *inheritance* dengan ketentuan sbb:

- Program menerima masukan dari *user*
- Program merupakan penerapan dari *multilevel hierarchy* (minimal 3 level)
- Program mengimplementasikan *overriding method*
- Program mengimplementasikan *keyword super*